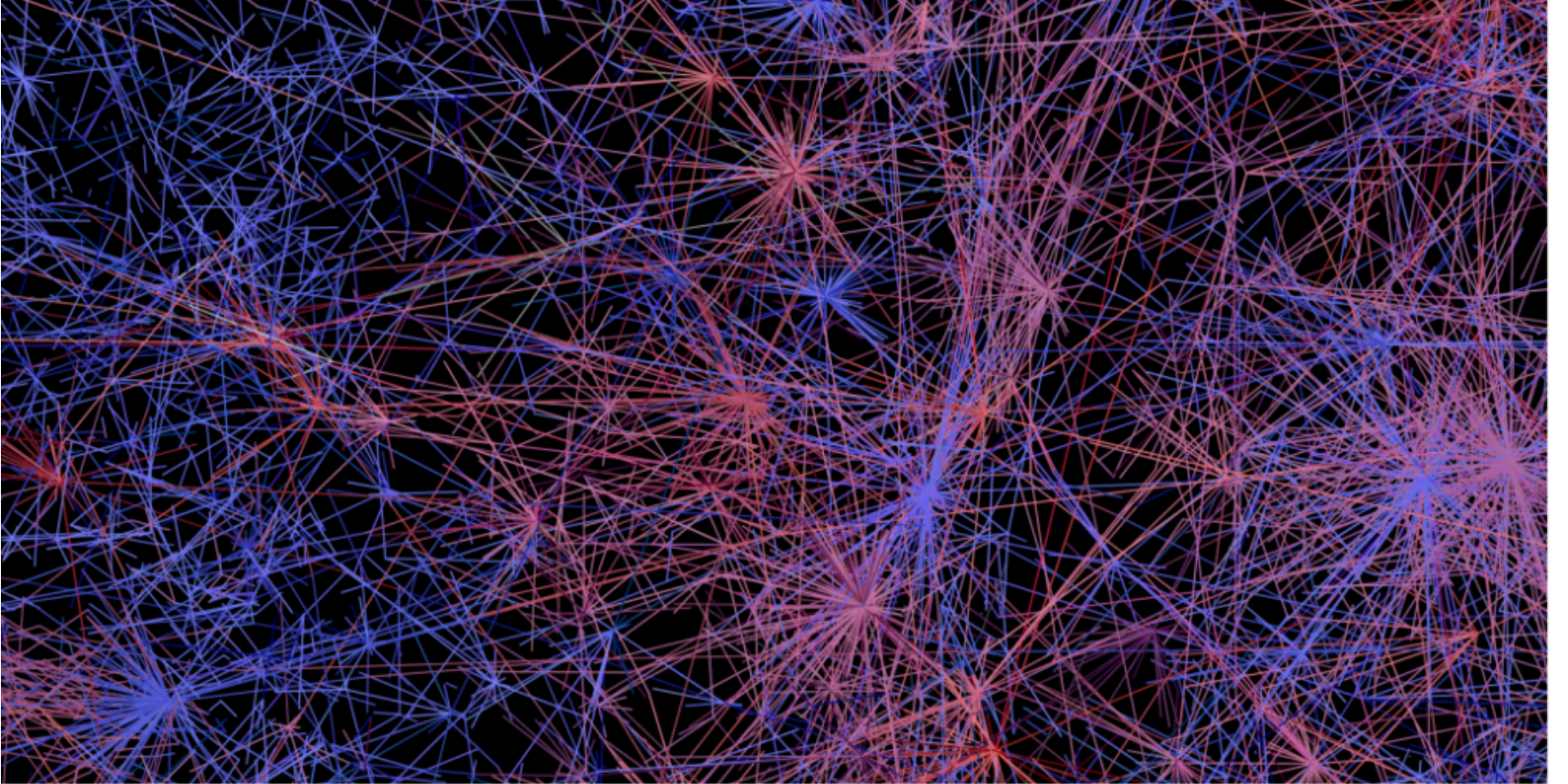


BİLGİSAYAR AĞLARI



KATMANLI MİMARİLERE GİRİŞ

- Başlangıçta ağların gelişimi çok karmaşıktı. Bilgisayar firmaları tarafından üretilen nic'lerin hepsi üretilen firmaya özeldi. Bu yüzden bir firma tarafından üretilen bir nic ile başka firma tarafından üretilen bir nic uyumsuzdu.

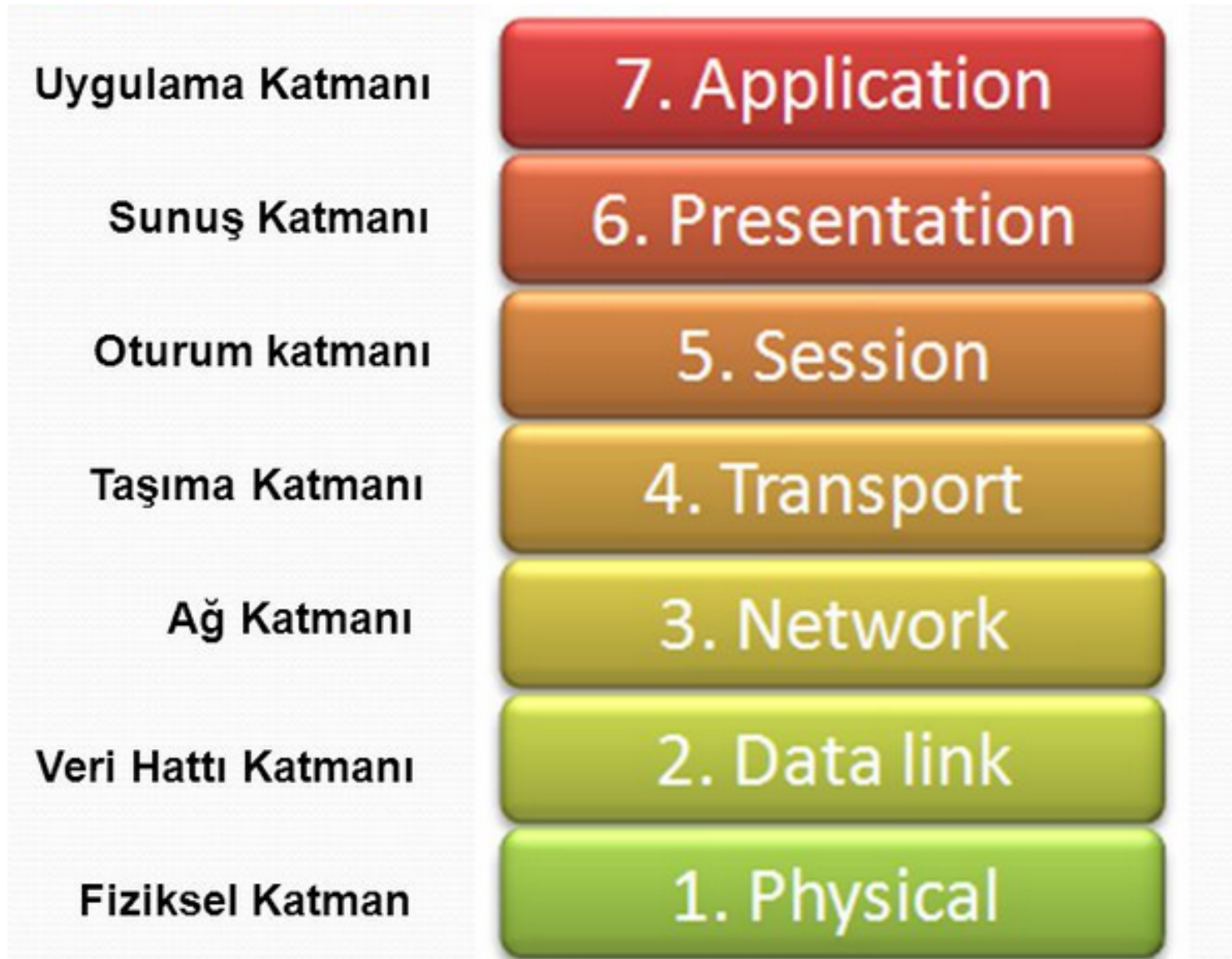
OSI MODELİ

- Burada ise OSI model adlı bir katmanlı mimari yaklaşımı geliştirildi. Bu yaklaşımda ise yazılımcıların ve donanım üretenlerin bu yaklaşımı baz alarak birşey üretmesi ile farklı firmaların ürettiği NIC'lerin veya yazılan uygulamaların birbiriyle haberleşmesi kolay kılındı. Bu yaklaşım 1984 yılında OSI(International Organization for Standardization) tarafından geliştirildi.

OSI MODELİ

- OSI sadece ağı kolaylaştıran bir model değildir. Bu model ile insanlara network hakkında birşeyler öğretmek veya üzerinde çalışmak kolaylaştırıldı.
- İlk dört katman ağ konusu hakkında çok önemlidir ve Networking olarak adlandırılır. Geri kalan 3 katman ise Application olarak adlandırılır.

OSI MODELİ



OSI MODELİ KATMANLARI

- Physical Layer : Bu katman voltaj seviyeleri, zamanlama, fiziksel veri hızları, fiziksel konektörler ve benzeri şeyleri tanımlar. Fiziksel olduğu için her şeye dokunabilirsiniz. Sinyalin dijital anlamının bozulmamasını sağlar. Ayrıca sinyal üzerindeki son değişiklik ve bilgi ekleme işlemlerini de yaptıktan sonra kablo üzerine gönderilme işlemini gerçekleştirir.

OSI MODELİ KATMANLARI

- Data Link: Bu katman verinin doğru bir şekilde biçimlendirilmesini sağlar. Hata tespiti ile ilgilenir ve Network katmanından alınan verinin doğru bir şekilde Physical katmana aktarılmasını sağlar. Burada Network katmanından alınan verinin hatalı olup olmadığını CRC (Cyclical Redundancy Check / Döngüsel Yedeklilik Kontrolü) işlemi yaparak kontrol eder. Eğer bir hata var ise burada Transport katmanı ile iletişime geçilir.

OSI MODELİ KATMANLARI

- Data link katman protokolleri
 - Logical Link Control (LLC) : LLC veri bağlantısı işlemini kontrol eden bir alt katmandır.
 - Media Access Control (MAC) : Bilgisayar içerisindeki fiziksel katmanın network kartına erişimini sağlar.

OSI MODELİ KATMANLARI

- Network :Bu katman bağlantıyı ve yol seçimini belirler.Ayrıca bu katman mantıksal adreslerin(MAC) fiziksel adreslere (IPv4 , IPv6) adreslerine çevirilmesini sağlar.Bu katmanda gönderilen verinin hedefe ulaşincaya kadar en iyi yolları seçerek gider.
- Network katman protokolleri
 - IP
 - ICMP
 - ARP
 - RARP

OSI MODELİ KATMANLARI

- Transport :Bu katman verinin alınmasına , iletilmesine ve hata kontrolleri ile ilgilenir.Bu katmanda alınan paketlerde bir eksiklik olup olmadığı , sırasının doğruluğu ve paketlerde herhangi bir hatanın olmadığını kontrol eder.Bu katmanda gönderilen veya alınan paketlerin eksikliği veya hatalı olması şeklinde tekrar gönderilmesi sağlanır.
- Transport katman protokolleri.
 - TCP : Verileri güvenli bir yol ile alan , gönderen bir protokoldür
 - UDP : Verileri güvensiz bir yol ile alan , gönderen bir protokoldür

OSI MODELİ KATMANLARI

- Session : Bu katman 2 host arasındaki bağlantının kurulması , yönetilmesi ve sonlandırılmasıyla ilgilenir. İnternet'teki bir web sitesine göz attığınızda, muhtemelen o web sitesini barındıran web sunucusunun tek kullanıcısı değilsinizdir. Bu web sunucusu tüm farklı "oturumları" takip etmelidir.

OSI MODELİ KATMANLARI

- Presentation : Bu katman, network içerisindeki bilgisayarlarda karşılıklı olarak iletişim halindeyken alınan verilerin okunabilmesi için verinin biçimine ve yapısının Application katmanı için uyumlu olmasını kontrol eder.Çoğu bilgisayarlar bir karakter yazma işleminde ASCII tablosunu kullanılır.İletişime geçilen bilgisayar farklı bir tablo kullanılıyorsa o bilgisayar için bir reformat yapılarak 2 bilgisayar arasındaki iletişim gerçekleştirilir

OSI MODELİ KATMANLARI

- Application : OSI modelin en yüksek kısmındaki katman application layer'dır. Uygulamaların network servislerine erişimini ve uygulamalar arası yönetimi sağlar. Bu katman doğrudan kullanıcı uygulamalarını çalıştıran servisleri içerir. Mesela; dosya transferi için, database'e erişim için, e-mail 'e erişimi için kullanılan yazılımlar. Application layer, genel network erişimini, akış kontrolünü ve hata kontrollerini tutar.

OSI MODELİ KATMANLARI

- OSI modeli üzerinde bulunan herhangi bir katmanı atlayarak geçemeyiniz. Application katmanından doğrudan network katmanına atlamamız mümkün değildir. Bir veriyi göndermek için bütün katmanları dolaşmamız gereklidir. Veri gönderimi için gerçek hayattan bir örnek'i inceleyelim

GERÇEK HAYATTAN BİR ÖRNEK

- Bilgisayarınız başında oturuyorsunuz ve bazı dosyaların internetten indirilmesi istiyorsunuz. Web tarayıcınızı açtınız ve indirmek istediğiniz bilgilerin bulunduğu web sitenin adresini url kısmına yazdınız. Bu sırada bilgisayarınız hedef siteye geçerli bir web sayfası almak istediği mesajını gönderir. Bu sırada application katmanında bulunan siz HTTP protokolünü kullanıyorsunuz.

GERÇEK HAYATTAN BİR ÖRNEK

- Presentation uygulamanın bilgisini belirli bir biçimde biçimlendirecektir.
- Session katmanı farklı oturumları birbirinden ayrılmasının kontrolünü yapacaktır
- Uygulamaya bağlı olarak, verileri web sunucusuna aktarmak için güvenilir bir (TCP) veya güvenilmez (UDP) protokol kullanılmaktadır. HTTP üzerinden veriler geleceği için güvenilir(TCP) bağlantı seçilecektir

GERÇEK HAYATTAN BİR ÖRNEK

- Bilgisayarınızın sahip olduğu unique(sadece size tek) ip adresinin'de içerildiği bir ip packet oluşturulacaktır.Bu paket içersinde Application , presentation ve session'dan gelen veri bilgileri kullanılacaktır.Bunun yanında ise transport katmanından gelen hangi veri taşıma protokol bilgisinin kullanılacağı gelecektir.Bunun yanında ise ip paketi kendi içersinde hedef ve kaynak ip adreslerini işleyecektir.

GERÇEK HAYATTAN BİR ÖRNEK

- Network katmanından ip packet bilgileri data link katmanında frame(çerçeve)'lere koyulacaktır.Fakat burada ise bilgisayarın ve paketi internet ortamına taşıyacak olan dış kapı(gateway)'in MAC adresleri konumlandırılacaktır.
- Son olarak ise herşey Physical katmanda 0,1(binary)'lere çevirilerek elektrik ile taşınacaktır.

```
▶ Frame 1: 74 bytes on wire (592 bits), 74 bytes captured (592 bits)
▶ Ethernet II, Src: AppleCom_67:49:3c (00:0a:95:67:49:3c), Dst: Kingston_2d:4a:a3 (00:c0:f0:2d:4a:a3)
▶ Internet Protocol, Src: 192.168.69.2 (192.168.69.2), Dst: 192.168.69.1 (192.168.69.1)
▶ Transmission Control Protocol, Src Port: 34059 (34059), Dst Port: http (80), Seq: 0, Len: 0
```

Here you see one of the Ethernet frames. Do you see the different layers of the OSI-model?

- Frame 1 / Ethernet II: This is the Data Link layer.
- Internet Protocol: This is the Network layer.
- Transmission Control Protocol: This is the Transport layer.

▼ Frame 4: 511 bytes on wire (4088 bits), 511 bytes captured (4088 bits)

Arrival Time: Oct 29, 2004 07:21:00.402698000 CEST

Epoch Time: 1099027260.402698000 seconds

[Time delta from previous captured frame: 0.000129000 seconds]

[Time delta from previous displayed frame: 0.000129000 seconds]

[Time since reference or first frame: 0.000282000 seconds]

Frame Number: 4

Frame Length: 511 bytes (4088 bits)

Capture Length: 511 bytes (4088 bits)

[Frame is marked: False]

[Frame is ignored: False]

[Protocols in frame: eth:ip:tcp:http]

[Coloring Rule Name: HTTP]

[Coloring Rule String: http || tcp.port == 80]

▼ Ethernet II, Src: AppleCom_67:49:3c (00:0a:95:67:49:3c), Dst: Kingston_2d:4a:a3 (00:c0:f0:2d:4a:a3)

► Destination: Kingston_2d:4a:a3 (00:c0:f0:2d:4a:a3)

► Source: AppleCom_67:49:3c (00:0a:95:67:49:3c)

Type: IP (0x0800)

▼ Internet Protocol, Src: 192.168.69.2 (192.168.69.2), Dst: 192.168.69.1 (192.168.69.1)

Version: 4

Header length: 20 bytes

► Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)

Total Length: 497

Identification: 0xf5db (62939)

► Flags: 0x02 (Don't Fragment)

Fragment offset: 0

Time to live: 64

Protocol: TCP (6)

► Header checksum: 0x37d7 [correct]

Source: 192.168.69.2 (192.168.69.2)

Destination: 192.168.69.1 (192.168.69.1)

▼ Transmission Control Protocol, Src Port: 34059 (34059), Dst Port: http (80), Seq: 1, Ack: 1, Len: 445

Source port: 34059 (34059)

Destination port: http (80)

[Stream index: 0]

Sequence number: 1 (relative sequence number)

[Next sequence number: 446 (relative sequence number)]

Acknowledgement number: 1 (relative ack number)

Header length: 32 bytes

► Flags: 0x18 (PSH, ACK)

Window size: 5888 (scaled)

► Checksum: 0x16ca [validation disabled]

► Options: (12 bytes)

► [SEQ/ACK analysis]

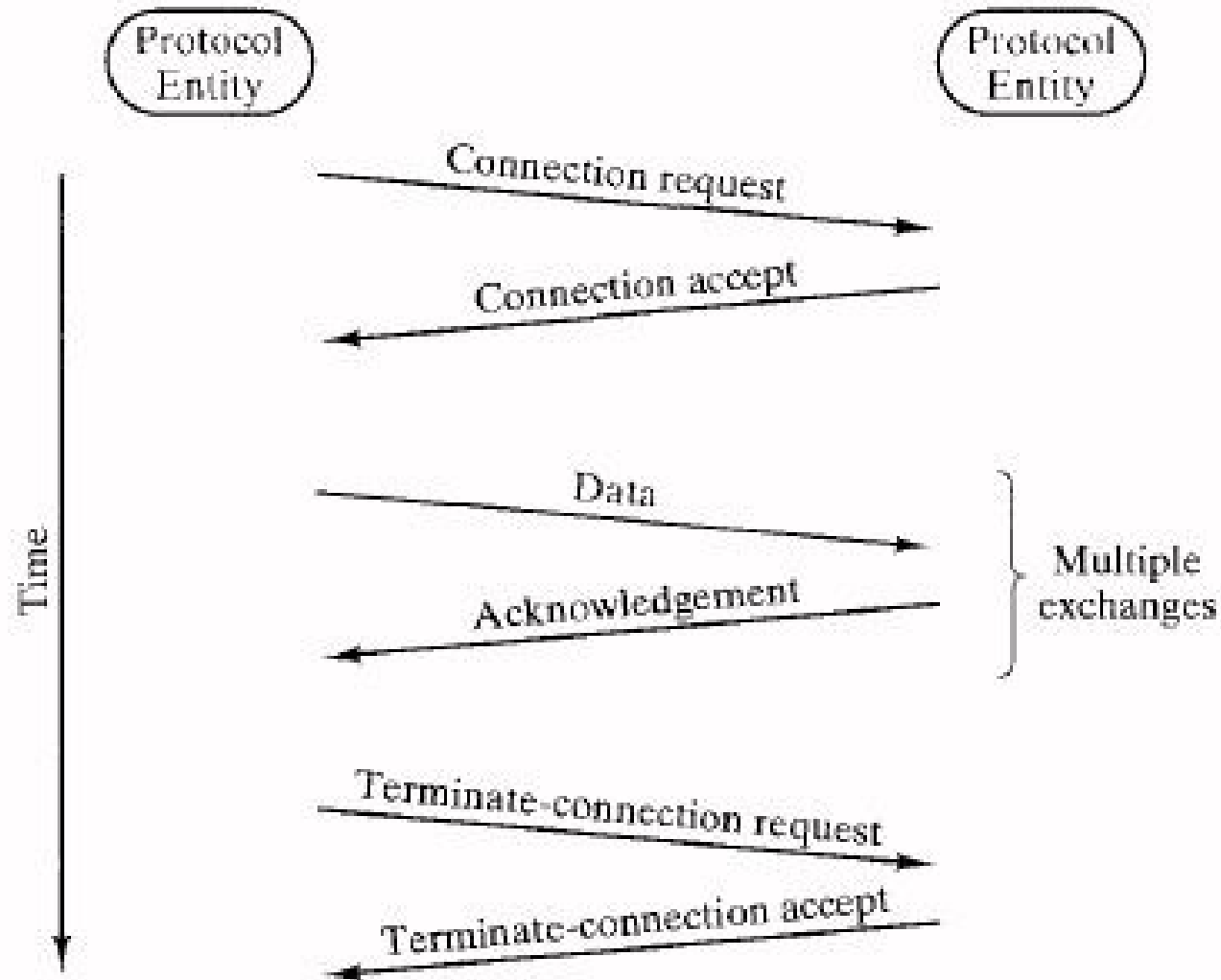
BİLİNMESİ GEREKENLER

- Protocol Data Unit(Protokol veri birimi)
 - Transport Katmanı : Burada oluşturulan veri topluluğuna **SEGMENT** denir.
 - Network Katmanı : Burada oluşturulan veri topluluğuna ise **PACKET** denir.
 - Data link Katmanı : Burada oluşturulan veri topluluğuna ise **FRAME** denir

BİLİNMESİ GEREKENLER

- Connection Oriented Communication (Bağlantı yönelimli iletişim) : Uzak bir bilgisayar ile iletişime geçtiğimizde , bu iletişimin doğru bir şekilde gerçekleşmesi için bazı durumların meydana gelmesi gereklidir.
 - Uzak makineye bağlantı istediği
 - Kabul edilen bağlantı isteğinden sonra verilerin gönderilmesi
 - Oturumun sonlandırılması

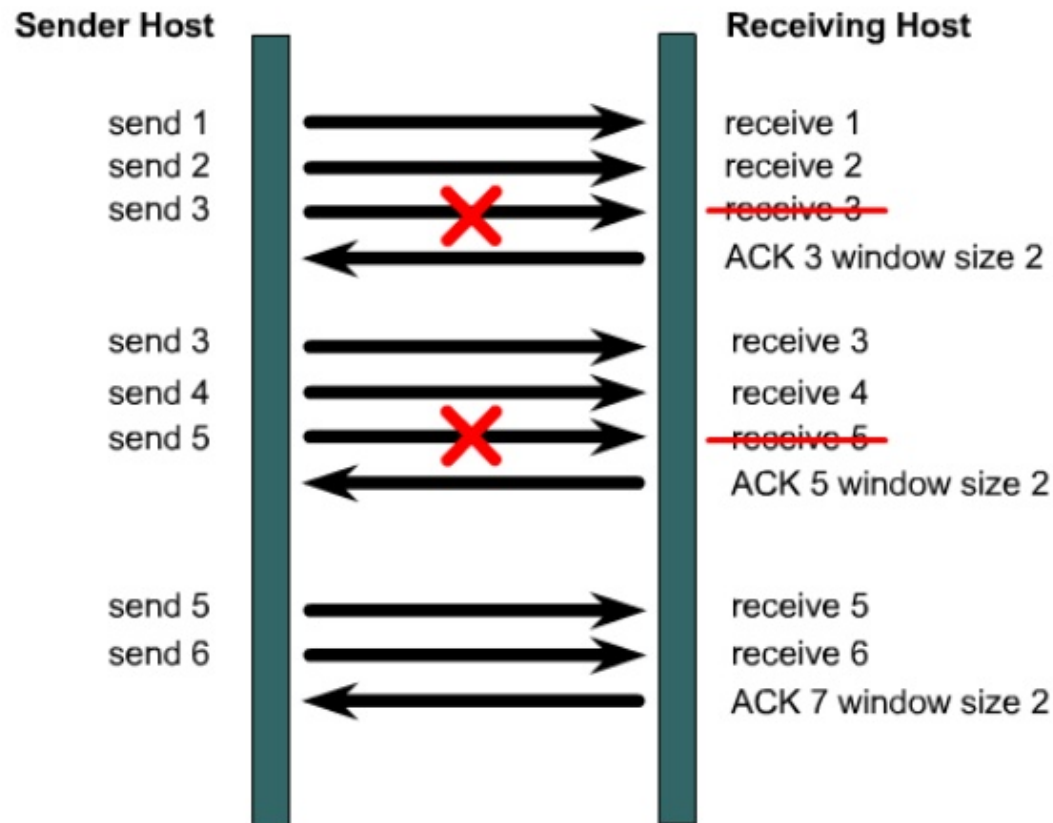
BİLİNMESİ GEREKENLER



BİLİNMESİ GEREKENLER

- Windowing : Gönderici ile Alıcı arasındaki TCP ile kurulan iletişimlerde göndericinin aynı anda kaç adet segment'in gönderileceği belirlenir. Bu işlemde Alıcı , Göndericinin kaç adet segment'ine cevap vermesi sonucunda ortaya çıkan sayıdır.

BİLİNMESİ GEREKENLER

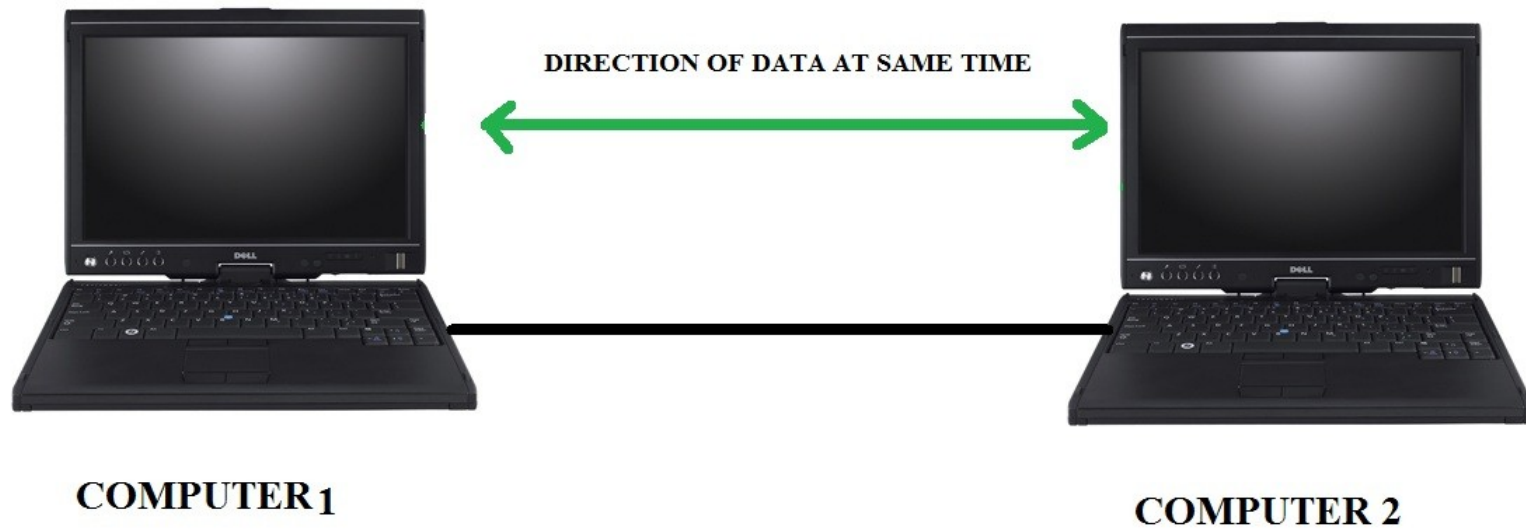


BİLİNMESİ GEREKENLER

- Full Duplex :Veri almayı ve göndermeyi eş zamanlı olarak gerçekleştirebilen iletişim türüdür. Bir istasyon aynı anda veri yollayıp alabilir. Noktadan noktaya iletişimlerde (point-to-point), 2. Katman olan Data-Link katmanı iletişimin half duplex (yarı çift yönlü) veya full duplex (tam çift yönlü) olup olmadığını belirler. Tam çift yönlü iletişimde de 2. katman medyanın iki ucundaki cihazların veri iletimi için uygunluğunu tanımlar. Tam çift yönlü haberleşmede collision (çakışma) olmaz

BİLİNMESİ GEREKENLER

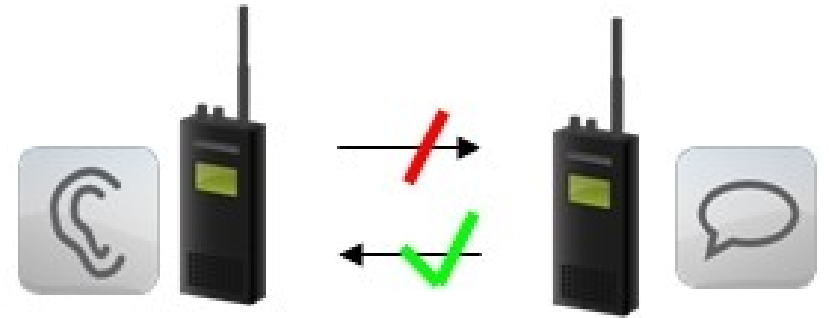
FULL DUPLEX TRANSMISSION MODE



BİLİNMESİ GEREKENLER

- Half Duplex : Bir istasyon veri yollarken diğer istasyonun sadece veri almasına izin veren iletişim türüdür. Yarı çift yönlü iletişimde cihazlar çift yönlü veri gönderip alabilirler; fakat bunu eş zamanlı olarak yapamazlar. Birden fazla istasyon aynı anda veri göndermeye çalıştığı zaman Ethernet protokolü devreye girerek çakışmayı önler.

BİLİNMESİ GEREKENLER



BİLİNMESİ GEREKENLER

